

30
CLAIMS

I claim:

1. A method for generating a schedule for a plurality of resources to complete a plurality of tasks, comprising the steps of:
- 5 receiving input information including:
- a resource calendar which identifies the plurality of resources and includes a set of resource constraints for each of the plurality of resources, and
- 10 a task list which identifies the plurality of tasks and includes a set of task constraints for each of the plurality of tasks, each of the set of task constraints identifying a plurality of assigned resources and a work-amount for each of the assigned resources;
- 15 generating assignments for each of the plurality of tasks, each of the assignments identifying a specific assigned resource, a specific work-amount required by the specific assigned resource, and a parent task, the parent task being one of the plurality of tasks and identifying the specific resource; and
- 20 scheduling each of the assignments in accordance with the set of resource constraints for the specific assigned resource and the set of task constraints for the parent task.

097244-1300

2. The method of Claim 1, wherein each of the sets of resource constraints identify at least one time-slot and an earliest available date for the resource, the time-slots indicating spans of time in which at least one of the assignments can be scheduled for the resource.

5 3. The method of Claim 1, wherein at least one of the sets of task constraints includes an assignment limit scheduling constraint for at least one of the plurality of assigned resources, the assignment limit scheduling constraint identifying a maximum duration that the assigned resource can be scheduled for the task associated with the set of
10 task constraints.

4. The method of Claim 1, wherein at least one of the sets of task constraints includes a must-finish-by scheduling constraint, the must-finish-by scheduling constraint identifying a date on which the task associated with the set of task constraints must be completed.

15 5. The method of Claim 1, wherein at least one of the sets of task constraints identifies a dependency scheduling constraint, the dependency scheduling constraint identifying at least one of the plurality of tasks that must be completed before the task associated with the set of task constraints can be started.

20 6. The method of Claim 1, wherein at least one of the sets of task constraints identifies a start-on-or-after scheduling constraint, the start-on-or-after scheduling constraint identifying a date on which the task associated with the set of task constraints must not start before.

5

10

15

20

20

25

rescheduling each of the assignments in accordance with the new set of resource constraints for the specific assigned resource and the task constraints for the parent task.

13. The method of Claim 1, further comprising the steps of:

receiving a new set of task constraints for at least one of the plurality of tasks; and

5 rescheduling each of the assignments in accordance with the resource constraints for the specific assigned resource and the new set of task constraints for the parent task.

14. The method of Claim 1, wherein each of the sets of resource constraints identify at least one available time-slot for the resource associated with the set of resource constraints, the time-slot identifying a span of time that the assignments can be scheduled for the resource, and the scheduling step further comprises the steps of:

selecting a first assignment identifying a first assigned resource;

15 associating the first assignment with a first available time-slot in the resource calendar for the first assigned resource;

selecting a next assignment identifying the first assigned resource and repeating steps b and c with the next assignment until all of the assignments identifying the first assigned resource have been scheduled;

20 and

selecting a next assigned resource and repeating steps a-d for the next assigned resource until all of the assignments for each of the plurality of resources have been scheduled.

0972141-112200

18. The method of Claim 17, wherein at least one of the assignments has a start-on scheduling constraint, and the generating step further comprises the step of assigning each of the assignments having the start-on scheduling constraint a higher priority-order than all other of the assignments.

19. The method of Claim 17, wherein each of the assignments has at least one weighted scheduling constraint, and the generating step further comprises the step of assigning the priority-order for each of the assignments as a function of the weighted scheduling constraints.

20. The method of Claim 16, wherein the ordering step further comprises the steps of:

placing each of the assignments into one of a plurality of groups, each of the assignments being placed into a particular group in accordance with a first function of the scheduling constraints and each of the plurality of groups assigned a priority in accordance with a second function of the scheduling constraints; and

for each of the plurality of groups, ordering each of the assignments in each of the groups in accordance with a specific function of the scheduling constraints associated with the group.

21. The method of Claim 20, wherein the scheduling constraints associated with the highest priority group comprise a start-on scheduling constraint.

097244-1300

22. The method of Claim 20, wherein each of the assignments are placed into at least one of four groups, and the placing step further comprises the steps of:

5 identifying a first group of the assignments consisting of each of the assignments having a start-on scheduling constraint;

identifying a second group of the assignments consisting of each of the assignments having a must-finish-by scheduling constraint and which are not included in the first group;

10 identifying a third group of assignments consisting of each of the assignments having no dependency scheduling constraints and which are not included in first group; and

identifying a fourth group of assignments comprising any of the assignments that are not included in one of the first and third groups.

15 23. The method of Claim 20, wherein the scheduling step further comprises the steps of:

scheduling each of the assignments in the highest priority group;

20 scheduling each of the assignments in the next highest priority group; and

repeating the scheduling steps until all of the assignments in each of the plurality of groups have been scheduled.

097244-1220
002277-14260

24. A computer-readable medium on which is stored a computer program for creating an assignment-oriented schedule, comprising the steps of:

- 5 a. receiving input information including:
a resource calendar which identifies resources,
available time-slots for each of the resources, and an earliest start date for
each of the resources, and
a task list which identifies tasks, each of the tasks
having a set of task constraints, a priority which is a function of the task
constraints, and a predecessor count, each of the sets of task constraints
10 identifying assigned resources for the task associated with the set of task
constraints and a work-amount for each of the assigned resources;
b. generating assignments for each of the tasks, each
of the assignments identifying a specific assigned resource, a specific work-
15 amount, and the task the assignment is generated from as a parent task;
c. ordering the assignments by:
1. identifying a first assignment group
comprising the assignments having a start-on scheduling constraint in the
set of task constraints of the parent task,
20 2. identifying a second assignment group
comprising the assignments having a must-finish-by scheduling constraint
in the set of task constraints of the parent task, and which are not identified
for the first assignment group,
3. identifying a third assignment group
25 comprising the assignments having a predecessor count with a value of
zero, and which are not identified for the first assignment group, and

d. scheduling each of the assignments in the first
5 assignment group into a time-slot for the assigned resource which
corresponds with the start-on scheduling constraint for the assignment; and

10 2. selecting a first assignment from the third
assignment group as a current assignment, the current assignment
identifying the current resource,

4. calculating a current end date based on all of the assignments that have been scheduled and the current assignment as though it has been scheduled,

25 6. scheduling the current assignment in the
first available time-slot for the current resource,

7. if all of the assignments having a common parent task have been scheduled:

moving all of the assignments in the fourth assignment group, which have the predecessor count of the parent task equal to the value of zero, to the third assignment group,

9. selecting a next resource as the current resource and repeating steps e2 through e9 until all of the assignments have been scheduled.

25. A computer system for creating an assignment-oriented schedule, comprising:

a processing unit;

a memory storage device;

5 an input device coupled to said processing unit for receiving information;

a pixel-based display device coupled to said processing unit for displaying data;

10 a program module, stored in the memory storage device, for providing instructions to the processing unit;

the processing unit, responsive to the instructions of the program module, operative to:

a. receive from the input device, input information including:

15 a resource calendar which identifies resources available for scheduling, available time-slots for each of the resources, and an earliest start date for each of said resources, and

20 a task list which identifies tasks to be performed, each of said tasks having task constraints, a priority-order which is a function of said task constraints and a predecessor count, said task constraints including a list of assigned resources for each of said tasks, a work-amount for each of said assigned resources, and at least one scheduling constraint;

25 b. generating assignments for each of the tasks, each of said assignments identifying a specific assigned resource, a specific work-amount, and a parent task;

0973141-11200

c. scheduling each of said assignments that have a parent task with a start-on scheduling constraint, into a time-slot of said resource calendar for said specific assigned resource, said time-slot corresponding with said start-on scheduling constraint;

5 d. scheduling each of said assignments not previously scheduled by:

1. selecting a first resource from the resources as a current resource;

10 2. if at least one unscheduled assignment exists which identifies said current resource as said specific assigned resource, has a parent task with a predecessor count equal to zero, and has a priority-order greater than all other unscheduled assignments which identify said current resource as said specific assigned resource and have a parent task with a predecessor count greater than zero:

15 selecting a current assignment from said unscheduled assignments which identifies a parent task having predecessor count equal to zero and a highest priority-order,

calculating a current end date based on all of said assignments that have been scheduled and said current assignment,

20 scheduling each of said assignments not previously scheduled which identify said current resource as said specific assigned resource, identify a parent task having a must-finish-by scheduling constraint that is earlier than said current end date and identify a parent task with a predecessor count equal to zero, in the first
25 available time-slot for said current resource,

scheduling said current assignment in the first available time-slot for said current resource,

00272441.11200

if all of said assignments identifying a common parent task have been scheduled, decrementing said predecessor counts for each of said plurality of tasks which have a dependency constraint identifying said common parent task as a predecessor task, and

5

repeating step d2;

3. if any of said assignments have not been scheduled, selecting a next resource from said plurality of resources as said current resource and repeating steps d2 and d3.

002211 141460

26. A method for generating a plurality of individually schedulable assignments for a task, based upon task constraints associated with said task, said task constraints identifying N resources assigned to said task where N is a positive integer, and a required work-amount
5 corresponding to each of said N resources, comprising the steps of:

dividing said task into N assignments, each of said N assignments being associated with one of said N resources; and

equating a work-amount for each of said N assignments to the required work-amount corresponding to said resource associated
10 with said assignment.

27. The method of Claim 26, wherein said task constraints identify one or more scheduling constraints and further comprising the step of associating each of said N assignments with said scheduling constraints.

28. The method of Claim 26, wherein said task constraints identify one or more scheduling constraints and further comprising the step of associating each of said N assignments with said task being divided.
15

002211 "T4T2260

sh
B

Sub
Q2

receiving a task description for said task, said task
description identifying N resources assigned to said task where N is a
5 positive integer, a required work-amount corresponding to each of said N
resources, and one or more scheduling constraints for said task:

10 equating a work-amount for each of said N assignments
to the required work-amount corresponding to said resource identified by
said assignment; and

15

DIVISION OF

30. A computer system for generating assignments for a task, comprising:

- a processing unit;
- a memory storage device;
- 5 a program module, stored in the memory storage device for providing instructions to the processing unit;
- the processing unit, responsive to the instructions of the program module, operative to:
 - 10 receive a task description for the task, the task description identifying N resources assigned to the task where N is a positive integer, and a total amount of required work for the task; and
 - divide the task into N assignments, each of the N assignments identifying one of the N resources.

31. The computer system of Claim 30, wherein the
15 processing unit is further operative to set a work-amount for each of the N assignments to the total amount of required work divided by N.

32. The computer system of Claim 30, wherein the task
description includes an assignment limit for at least one of the N
resources, and the processing unit is further operative to set a work-
20 amount for each of the N assignments in accordance with the assignment limits and in a manner that the summation of all of the work-amounts is equal to the total amount of required work.

09721141-112200

shy
B1

33. The computer system of Claim 30, wherein the task description includes one or more scheduling constraints for the task, and the processing unit is further operative set a work-amount for each of the N assignments as a function of the scheduling constraints and in a manner that the summation of all of the work-amounts is equal to the total amount of required work.

34. The computer system of Claim 30, wherein the task description includes one or more scheduling constraints for the task, and the processing unit is further operative to associate each of the N assignments with the scheduling constraints.

35. The computer system of Claim 30, wherein the task description includes one or more scheduling constraints for the task, and the processing unit is further operative to:

- associate each of the N assignments with the scheduling constraints; and
- assign a priority to each of the assignments as a function of the scheduling constraints.

add
p2

002211-112200

36. A method for scheduling an ordered list of assignments, comprising the steps of:

receiving a resource calendar which identifies unscheduled time-slots available resources;

5 receiving the ordered list of assignments, each of the assignments identifying at least one of the available resources; and

scheduling each of the assignments in the ordered list of assignments into a next unscheduled time-slot on the resource calendar for the available resource identified by the assignment.

10 37. The method of Claim 36, wherein the unscheduled time-slots and the assignments have a duration, and the scheduling step further comprises the steps of:

if the duration of the assignment is longer than the duration of the next unscheduled time-slot, sub-dividing the assignment and
15 scheduling each sub-division of the assignment separately in the next unscheduled time-slot;

if the duration of the assignment is not longer than the duration of the next unscheduled time-slot, schedule the assignment in the next unscheduled time-slot.

20 38. The method of Claim 36, wherein the resource calendar further identifies an earliest available start date for the available resources, and the scheduling step further comprises the step of selecting the first unscheduled time-slot that is after the earliest available start date as the next unscheduled time-slot .

09723144 112200

39. A computer-readable medium on which is stored a computer program for generating a schedule for a plurality of unscheduled assignments, each of the unscheduled assignments identifying at least one of a plurality of resources, a work-amount, a priority-order, and one of a plurality of parent tasks, each of the resources having a resource calendar identifying available time-slots for the resource, and each of the plurality of parent tasks having at least one scheduling constraint and a predecessor count, comprising the steps of:

- a. selecting a current resource from the plurality of resources;
- b. selecting a current assignment from the plurality of unscheduled assignments, the current assignment identifying the current resource, identifying a parent task having a start-on scheduling constraint, and having a highest priority-order;
- c. scheduling the current assignment in the first available time-slot of the resource calendar for the current resource;
- d. identifying the current assignment as scheduled and repeating steps b-d if additional unscheduled assignments identifying the current resource are remaining;
- e. if additional unscheduled assignments which identify a parent task having a start-on scheduling constraint exist, selecting a next current resource and repeating steps a-e with the next current resource.
- f. scheduling each of the remaining unscheduled assignments by:
 1. selecting a next resource from the plurality of resources as the current source;

00221144 260

2. if at least one qualifying unscheduled assignment exists, the qualifying unscheduled assignments must identify the current resource, identify a parent task having a predecessor count equal to zero, and have a priority-order greater than any other unscheduled assignments which identify the current resource and identify a parent task having a predecessor count greater than zero:

selecting a qualifying unscheduled assignment with the greatest priority-order as the current assignment, calculating a current end date based on all of the assignments that have been scheduled and the current assignment, if any unscheduled assignments, which identify the current resource and identify a parent task having a must-finish-by scheduling constraint that is earlier than the current end date, scheduling each such assignment into the first available time-slot for the current resource, otherwise, scheduling the current assignment in the first available time-slot for the current resource,

if all of the assignments identifying a common parent task have been scheduled, decrementing the predecessor counts for each of the plurality of parent tasks which have a dependency scheduling constraint identifying the common parent task as a predecessor task, and repeating step f2;

3. if additional unscheduled assignments exists, selecting a next resource from the plurality of resources as the current resource and repeating steps f2 and f3.